Research Notes: AI-Generated Object Affordances for Games

The Core Idea

Teaching AI to understand what you can DO with 3D objects in games, not just what they look like.

The Problem:

- Right now, game devs manually program every single interaction (sit on chair, throw rock, hide behind wall)
- Takes forever, limits creativity, blocks smaller studios from competing

My Solution: AI looks at a 3D model → understands shape + material → automatically suggests realistic interactions

Example:

- Input: 3D chair model
- AI recognizes: flat surface (wood)
- Output: sit, stand on, throw, use as fuel, hide behind

Why This Matters

Game Industry Reality:

- $200B+ industry where interaction design is a MASSIVE bottleneck
- 80% of dev time = content creation, not actual gameplay innovation
- Indie devs can't match AAA richness because they don't have resources

What I'm Building: First system that automatically bridges the gap between "here's a 3D object" and "here's what players can do with it"

Impact:

- Cut development costs 30-50% for interaction-heavy games
- Smaller teams → AAA-quality experiences

- Opens up totally new game genres based on rich object interaction

The Gap We Fill

What exists:

- MeshyAI, StabilityAI → generate 3D models from text/images
- They make objects, but don't understand PURPOSE

What's missing: Intelligence layer that says "this chair affords sitting, throwing, hiding"

My contribution: Not competing on model generation. Building the NEXT layer—making generated models actually useful for interactive stuff.

Research Questions

Primary

1. Geometry → Affordance Mapping How can AI analyze 3D mesh to identify interaction surfaces automatically?

2. Material-Based Prediction How does AI classify materials from textures and predict physical interactions (breakable, burnable, throwable)?

3. Context-Aware Generation How does AI adapt affordances based on game genre, setting, player capabilities?

Secondary

- Affordance completeness: catching all obvious + creative uses
- Cross-object interactions: chair + table = workspace setup?
- Implementation translation: auto-generate Unity/Unreal scripts
- Quality assessment: does this feel natural to players?

Technical Challenges

Data Problems:

- Limited 3D models with affordance labels

- Inconsistent formats (FBX, OBJ, glTF)
- No ground truth data from devs

3D Processing:

- Efficient mesh analysis for real-time
- Handling different coordinate systems
- Scaling across vastly different object sizes

Developer Integration:

- Works with Unity, Unreal, Godot
- Generates usable scripts devs can actually implement
- Plays nice with existing physics systems

Generalization:

- Works on novel objects (not just training data)
- Handles stylized art (cartoon, abstract)
- Adapts to different art styles

Research Challenges

- Combine geometry + material/texture analysis
- Understand object semantics from visual 3D data only
- Handle decorative vs functional objects

Context Awareness:

- Adapt to game genres (horror vs puzzle vs sim)
- Cultural/temporal contexts (medieval vs space-age chair)
- Balance realism with gameplay needs

Validation:

- Does this feel "natural" to players?
- Measuring completeness
- Benchmarking against human annotations

Performance:

- Process high-poly models without lag
- Cache similar objects
- Optimize for indie devs with limited compute